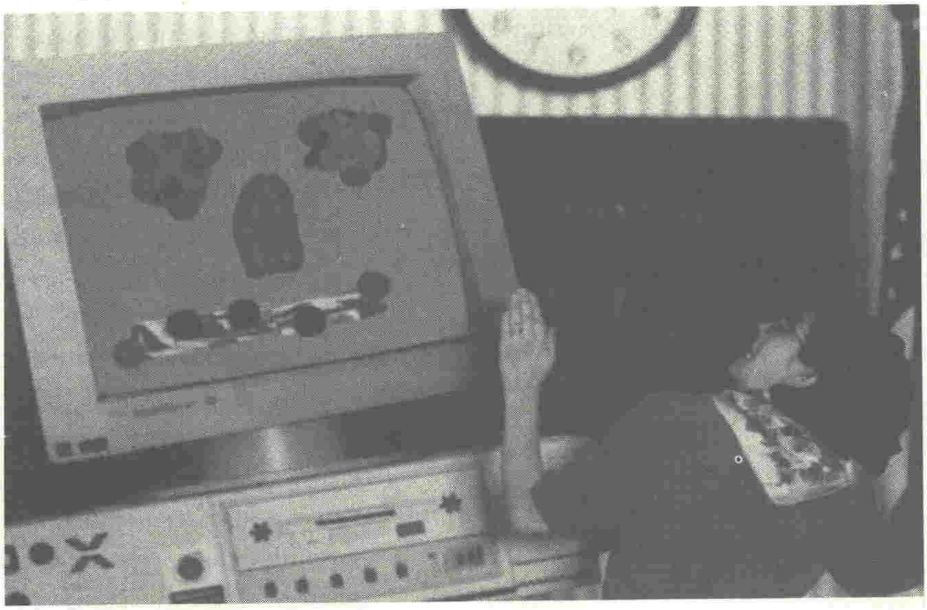
ขจร โรจนเมธินทร์

คุณรู้จัก MMX หรือยัง ?



MMX จะช่วยเพิ่มประสิทธิภาพงานทางด้านมัลติมีเดียของระบบที่ใช้ X86

โปรเซสเซอร์ที่ใช้เทคโนโลยี x86 ส่วนใหญ่มีประสิทธิภาพด้านมัลติมีเดียที่ต่ำกว่า โดยเฉพาะเมื่อเปรียบเทียบกับเครื่องที่ใช้เทคโนโลยี RISC กลุ่มบริษัท Sun, SGI และ Apple ดังนั้นอินเทลและผู้ผลิตโปรเซสเซอร์ตระกูล x86 รายอื่นจึงร่วมมือกันนำเทคโนโลยี MMX เข้ามาใช้เพื่อหวังจะแก้ปัญหาดังกล่าว

อินเทลอ้างว่า MMX จะช่วยเพิ่มประสิทธิภาพในการทำงานของโปรเซสเซอร์ตระกูล x86 กับแอพพลิเคชันบางประเภท โดย MMX จะเพิ่มประสิทธิภาพในการทำงานด้านกราฟิก ซาวนด์ และแอพพลิเคชันที่ทำงานด้านสื่อสารขึ้นเกือบ 400 เปอร์เซ็นต์ อินเทลกล่าวว่าแอพพลิเคชันด้านมัลติมีเดียจะทำงานได้เร็วขึ้น 50 -200 เปอร์เซ็นต์

MMX คือกลุ่มคำสั่งใหม่ของซีพียู ซึ่งมีเพิ่มมาอีก 57 คำสั่ง คำสั่งชุดนี้จะช่วยเพิ่มความเร็วในการคำนวณที่ต้องใช้บ่อย ๆ ในมัลติมีเดียแอพพลิเคชัน โดยเฉพาะแอพพลิเคชันที่ต้องทำงานเป็นลูปขนาดใหญ่จะได้รับประโยชน์มาก อินเทลเชื่อว่ามัลติมีเดียแอพพลิเคชันส่วนใหญ่จะทำงานแบบอัลกอริธึมที่ซ้ำ ๆ กันเพื่อคำนวณข้อมูลขนาด 8 บิต หรือ 16 บิตเท่านั้น อินเทลกล่าวว่า MMX ไม่ได้ย่อมาจาก Multimedia Extensions แต่เป็นชื่อรหัสที่ใช้แทนชุดคำสั่งเท่านั้น สำหรับ MMX นี้เป็นคำสั่งที่เร่งความเร็วโดยใช้เทคนิคการประมวลผลแบบขนาน

MMX ไม่ได้เป็นโปรเซสเซอร์รุ่นใหม่ แต่เป็นชุดคำสั่งที่จะถูกเพิ่มเติมให้กับโปรเซสเซอร์เพนเทียมและเพนเทียมโปร โปรเซสเซอร์ตัวแรกที่สามารถใช้ MMX ได้ใช้รหัส P55C จะออกวางตลาดได้เป็นครั้งแรกในออสเตรเลียในช่วงกลางปี 1997 ส่วนเพนเทียมโอเวอร์ไดรฟ์ที่ใช้ MMX ได้จะออกมาภายหลัง และเพนเทียมโปรที่ใช้เทคโนโลยี MMX อินเทลจะวางตลาดในอเมริกาในต้นปี 1997

MMX ทำงานอย่างไร

MMX ทำงานผ่านวิธีการที่เรียกว่า Single Instruction, Multiple Data (SIMD) ซึ่งยอมให้ข้อมูลหลาย ๆ ชุดถูกนำมาประมวลผลขนานกันไป โดยพื้นฐานแล้ว MMX จะยอมให้การประมวลผลข้อมูลในลักษณะของแบตช์มากกว่าการทำงานแบบเดี่ยว ๆ MMX จะใช้รีจิสเตอร์ขนาด 64 บิตเพื่อเก็บข้อมูล ดังนั้นโปรแกรมจะเก็บข้อมูลขนาด 8 บิตต่อไบต์ ไว้ได้ถึง 8 ชุด ข้อมูลขนาด 16 บิตต่อเวิร์ด ได้ 4 ชุด ข้อมูลขนาด 32 บิตต่อดับเบิลเวิร์ด ได้ 2 ชุด และข้อมูลขนาด 64 บิต ได้ 1 ชุด ซึ่งเมื่อเราอัดข้อมูลไว้ในรีจิสเตอร์มากกว่า 1 ชุด โปรเซสเซอร์ที่มี MMX จึงสามารถประมวลผลแบบขนานได้ด้วยคำสั่งเดียว ไม่ว่าจะเป็นการบวก สะสม เปรียบเทียบ หรือโอเปอเรชันอื่น ๆ จะสามารถทำงานแบบแบตช์กับข้อมูล 2, 4, 8 ชุดได้พร้อมกัน ประสิทธิภาพจะเพิ่มขึ้นอย่างเห็นได้ชัด โดยเฉพาะเมื่อข้อมูลที่ใช้งานมีขนาดเล็ก ๆ เพราะจำนวนคำสั่งจะลดลงประมาณ 1 ใน 8 จากเดิม

MMX จะช่วยเพิ่มความเร็วกับการทำงานแบบลูปจำนวนมาก ๆ ซึ่งใช้มากในมัลติมีเดียแอพพลิเคชัน หรือในกราฟิก

แอพพลิเคชันที่ต้องคำนวณมาก ๆ ซีพียูจะต้องประมวลผลค่าแต่ละพิกเซลด้วยคำสั่งหนึ่งคำสั่ง แต่ถ้าใช้ MMX ซีพียูจะประมวลผลพิกเซลได้ 2, 4, 8 พิกเซลด้วยคำสั่งเดียว ชุดคำสั่งใหม่ประกอบด้วยคำสั่งที่เกี่ยวกับคำสั่งในรีจิสเตอร์ MMX คำสั่งเหล่านี้จะใช้สำหรับการคำนวณพื้นฐานกับข้อมูลที่อัดอยู่ในรีจิสเตอร์ ไม่ว่าจะเป็นการเปรียบเทียบข้อมูลแต่ละตัว การแปลงค่าบิตในข้อมูล ทำโอเปอเรชันทางตรรก (AND/OR/NOT) และเลื่อนข้อมูลแต่ละบิต

MMX มีรีจิสเตอร์ขนาด 64 บิต อยู่ 8 ตัว รีจิสเตอร์ใหม่จะถูกตั้งชื่อให้เหมือนกับรีจิสเตอร์ที่มีใน Floating-point unit นั่นหมายถึงระบบปฏิบัติการสามารถติดต่อกับรีจิสเตอร์โฟลติ้งพอยน์ได้เช่นเดียวกับรีจิสเตอร์ MMX ทำให้ไม่ต้องไปออกแบบระบบ

ปฏิบัติการใหม่ นอกจากนี้โปรแกรมที่สร้างก่อนที่จะมี MMX ใช้ยังสามารถนำมาทำงานได้ดีบนโปรเซสเซอร์ MMX

ปัญหาอยู่ที่ซอฟต์แวร์

ปัญหาสำคัญของ MMX คือโปรแกรมที่เรียกใช้ชุดคำสั่ง MMX จะใช้ประโยชน์จากเทคโนโลยีการประมวลผลแบบใหม่นี้ได้เท่านั้น ประสิทธิภาพของเกมที่คุณชื่นชอบจะไม่เพิ่มขึ้นอย่างชัดเจนหากคุณซื้อโปรเซสเซอร์ MMX มาใช้ นอกเสียจากว่าเกมนั้นสามารถเรียกใช้ความสามารถของ MMX มาใช้ได้โดยตรง

แม้ว่า MMX จะแพร่หลายแล้วก็ตาม เราจะยังไม่เห็นแอพพลิเคชันที่เรียกใช้ MMX มีมากนัก อย่างไรก็ตาม ผู้ขายส่วนใหญ่จะบอกว่าซอฟต์แวร์ทำงานได้ทั้งโปรเซสเซอร์ที่มีและไม่มี MMX แทนที่จะจำกัดตัวเอง ทั้งนี้ซอฟต์แวร์สามารถตรวจสอบได้เองว่ามี MMX อยู่หรือไม่ และสามารถเลือกใช้เทคนิคการประมวลผลแบบอนุกรมหรือขนานได้

ขณะที่การพัฒนาโปรเซสเซอร์ที่มี MMX ยังไม่แน่นอน แต่ผู้ผลิตซอฟต์แวร์ส่วนใหญ่เริ่มหันมาสนับสนุนการใช้ชุดคำสั่ง MMX แล้ว เช่น คอมพายเลอร์บางตัว ซอฟต์แวร์ทางด้านเดสค์ท็อปพับลิชชิง มัลติมีเดียแอพพลิเคชัน และเพื่อเป็นการแนะนำการใช้เทคโนโลยี MMX อินเทลได้แนะนำ Internet Media Developer Support Program เพื่อที่จะแนะนำการพัฒนาสำหรับอินเทอร์เน็ต, DVD และ MMX

ไมโครซอฟท์ประกาศว่าผลิตภัณฑ์หลายตัวของตนสนับสนุนการใช้ MMX เช่น Direct 3D, ActiveMovie APIs นอกจากนี้ Visual C++ ตั้งแต่เวอร์ชัน 4.1 เป็นต้นไป สามารถคอมพายล์ให้โค้ดสามารถเรียกใช้ประโยชน์จาก MMX ได้อย่างเต็มที่ด้วย Watcom ก็ประกาศที่จะสนับสนุน MMX ในคอมพายเลอร์ C++ เวอร์ชันใหม่ที่จะออกมา บอร์แลนด์เองก็พร้อมจะสนับสนุน MMX ในเทอร์โบคอมพายเลอร์ทั้ง Pascal และ C++ รวมทั้ง Symantec ด้วยที่จะสนับสนุน MMX ในคอมพายเลอร์ C++ ด้วย นอกจากนี้ยังมีอีกหลายบริษัทที่พร้อมจะสนับสนุน MMX ในอนาคต เช่น Macromedia Director, Adobe Photoshop, CompCore's AC-3 software และ Crieterion's Renderware 3D เป็นต้น ส่วนไอบีเอ็มก็ใช้ MMX ใน VoiceType และ VoxWare ก็นำ MMX มาใช้ในการประมวลผลด้านเสียงเช่นกัน

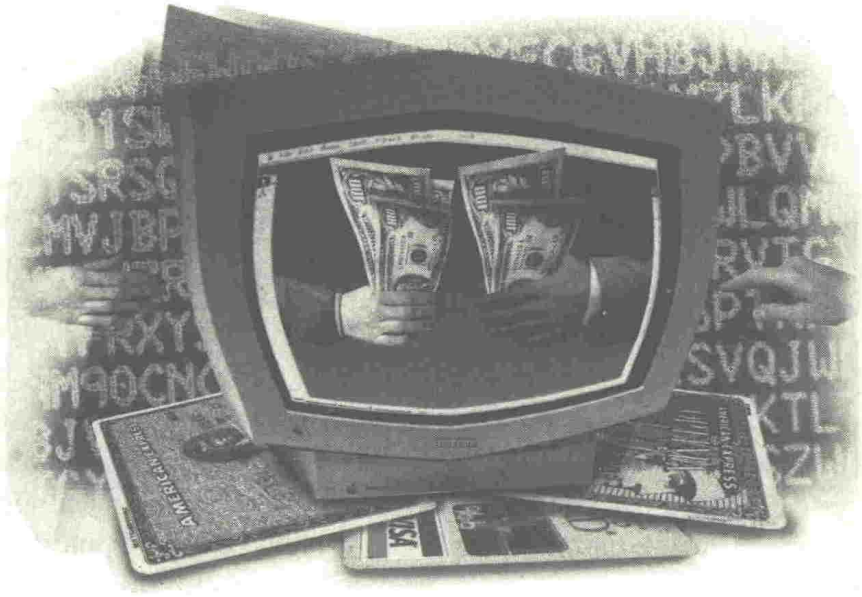
RISC ยังเป็นคู่แข่ง

เมื่อมีการเปิดตัวคำสั่ง MMX ในชิพของอินเทลนั้นอาจเป็นการปลุกบริษัทผู้ผลิตฮาร์ดแวร์รายอื่นให้หันมาสนใจบ้างก็ได้ ผู้ที่ได้รับผลกระทบจาก MMX โดยตรงก็ดูจะเป็นคู่แข่งของอินเทลที่สร้างเกม เดสค์ท็อปพับลิชชิง ซีเอดี ดิจิตอลเอนิเมชัน นอกจากนี้ก็คงเป็นผู้ผลิตเครื่อง RISC จาก Sun, Silicon Graphics เนื่องจากผู้บริโภคอาจหันใช้เทคโนโลยีของอินเทลที่มีราคาถูกกว่า สำหรับผู้ที่ใช้ Cyrix, AMD ก็ไม่ต้องตกใจ เพราะ AMD ได้รับลิขสิทธิ์เทคโนโลยี MMX จากอินเทล และจะมีใช้เป็นครั้งแรกในโปรเซสเซอร์ K6 แต่ Cyrix ไม่ได้รับลิขสิทธิ์การใช้เทคโนโลยีใหม่ แต่โปรเซสเซอร์ M2 ซึ่งเป็นคู่แข่งของเพนเทียม โปรมีคำสั่งไบนารีที่เข้าได้กับชุดคำสั่งใหม่ ทำให้ใช้แอพพลิเคชัน MMX ยังทำงานได้ ส่วนผู้ผลิตโปรเซสเซอร์รายอื่นยังไม่สามารถใช้ประโยชน์จากเทคโนโลยีใหม่นี้ ไม่ว่าจะเป็น PowerPC ของโมโตโรล่า, MIPS และ Alpha ของ Digital แม้ว่าสนับสนุนคำสั่งการประมวลผลแบบขนานก็ตาม ดังนั้นอินเทลจึงดูได้เปรียบกว่าสำหรับแอพพลิเคชันที่ทำงานมัลติมีเดีย

อย่างไรก็ตาม ทาง Silicon Graphics กลับไม่เห็นว่าเทคโนโลยีของอินเทลเป็นสิ่งที่น่ากลัวสำหรับเวอร์กสเตชันคุณภาพสูงและเซอร์ฟเวอร์ เพราะเชื่อว่าคงไม่มีผู้ผลิตพีซีรายใดที่จะสร้างเครื่องที่มีความสามารถเอนกประสงค์ไว้ในตัวเดียวกัน ผู้ใช้ที่ต้องทำงานงานซับซ้อนยังคงต้องการเครื่องที่มีประสิทธิภาพระดับเวอร์กสเตชันอยู่ดี ทางแอปเปิ้ลเองซึ่งเป็นผู้นำด้านเดสค์ท็อปพับลิชชิงก็คิดว่าถ้า MMX ทำงานได้อย่างจริงจัง ก็คงเป็นเรื่องที่น่ากลัว เพราะ Adobe Photoshop จะทำงานได้เร็วขึ้น 400 เปอร์เซ็นต์ บนโปรเซสเซอร์ MMX ตามตัวเลขของอินเทล แต่แอปเปิ้ลเชื่อว่าคงต้องใช้เวลาไม่ต่ำกว่า 18 เดือน จึงจะมีแอพพลิเคชันที่สนับสนุน MMX ออกมาให้ใช้ ทางด้านอินเทลเองก็เชื่อมั่นว่า MMX ไม่ได้มีประโยชน์เฉพาะแต่เกมเท่านั้น MMX จะมีประโยชน์โดยเฉพาะจะทำให้พีซีที่ใช้เพนเทียมโปรซึ่งมี MMX สามารถทำงานในลักษณะเวอร์กสเตชันอย่างเต็มรูปแบบ นอกจากนี้ยังมั่นใจว่าจะมีผู้ผลิตซอฟต์แวร์ที่สนับสนุนการใช้ MMX ออกมาแน่นอนในระยะเวลาอันสั้น คงไม่ต้องสงสัยเลยว่า MMX จะช่วยเพิ่มประสิทธิภาพงานทางด้านมัลติมีเดียของระบบที่ใช้ x86 และจะเป็นทางเลือกใหม่สำหรับผู้ใช้ที่ต้องประสิทธิภาพที่เพิ่มขึ้น

ดร.ปัญญา เปรมปรีดิ์

ประโยชน์ของการจัดเรียง



ความสามารถในการใช้ประโยชน์ของการจัดเรียงข้อมูลแบบ off-line จากเครื่องคอมพิวเตอร์

เครื่องมือที่สำคัญเป็นอันดับหนึ่งของนักออกแบบระบบงานคอมพิวเตอร์ก็คือ **ความสามารถในการใช้ประโยชน์ของการจัดเรียงข้อมูล** นักคอมพิวเตอร์รุ่นเก่า ๆ นั้นจะเข้าใจในเรื่องนี้เป็นอย่างดี ทั้งนี้เพราะโปรแกรมประเภทที่ทำงานแบบแบตช์โปรเซสซิงนั้นต้องใช้เทคนิคเหล่านี้ แต่นักคอมพิวเตอร์รุ่นใหม่นั้นไม่เคยผ่านงานประเภทนี้ งานของคนรุ่นใหม่เป็นแบบออนไลน์ และมีซิสเต็มส์ซอฟต์แวร์ดี ๆ เข้าช่วย เช่น ภาษาฐานข้อมูล คนรุ่นใหม่จึงมักจะไม่ซาบซึ้งกับเรื่องนี้ และเมื่อจะต้องออกแบบระบบงานที่มีข้อมูลแยะ ๆ ก็อาจออกแบบผิดพลาดได้ วันนี้ผมอยากจะหยิบยกตัวอย่างมาให้ดูสัก 2-3 เรื่อง และอยากจะแนะนำให้พวกเราที่อยากจะออกแบบระบบให้ดี ๆ ได้ใช้เวลาศึกษาเรื่องนี้เพิ่มอีกสักหน่อย

ผมจะขอยกตัวอย่างเรื่องที่เพิ่งจะเกิดขึ้นมาให้ดูก่อน เรื่องนี้เกี่ยวกับงานกาญจนาภิเษก คือธนาคารแห่งประเทศไทยได้เสนอให้พิมพ์ธนบัตรรุ่นพิเศษขึ้นมา 1 รุ่น รุ่นนี้จะใช้กระดาษชนิดพิเศษคือ ทำด้วยสารโพรลีเมอร์ กระดาษพิเศษนี้จะมีความมันเป็นเงางาม พิมพ์แล้วได้ภาพที่ชัดเจนและสวยงาม และสามารถใช้งานได้ทนทาน (แต่ส่วนใหญ่ประชาชนก็คงจะเก็บไว้เป็นที่ระลึกมากกว่า) นอกจากนี้แล้วตัวลายน้ำก็จะเปลี่ยนมาเป็นภาพประเภทสีเหลืองและเปลี่ยนไปตามมุมมอง นอกจากความพิเศษต่าง ๆ เหล่านี้แล้วก็ยังพิมพ์เป็นจำนวนจำกัด คือเดิมนั้นจะพิมพ์เพียง 5 แสนฉบับ แต่หลังจากที่สั่งพิมพ์ไปแล้ว และได้ซาวนด์เสียงดูในตลาดนักสะสมของที่ระลึกก็พบว่าต้องสั่งพิมพ์เพิ่มอีก 5 แสนฉบับ ธนบัตรเหล่านี้จึงมีหมายกำหนดที่จะส่งมอบเป็น 2 รอบ รอบแรกได้ส่งมาถึงเมืองไทยแล้ว ส่วนอีก 5 แสนฉบับรอบที่ 2 นั้นจะส่งมาประมาณเดือนตุลาคมหรือพฤศจิกายน 2539 นี้

ทีนี้ก็มาถึงลักษณะของโจทย์ที่จะให้พวกเรานักคอมพิวเตอร์ได้คิดกัน โจทย์บอกว่าธนาคารต้องการให้ธนบัตรรุ่นนี้ได้กระจายไปถึงมือประชาชนโดยทั่วถึง **ดังนั้นจึงกำหนดว่าประชาชนแต่ละคนจะซื้อได้ไม่เกินคนละ 5 ฉบับ** ทั้งนี้โดยในวันซื้อ (จะเริ่มเปิดให้ซื้อได้ในวันที่ 9 กรกฎาคม 2539) นั้นให้จ่ายเงินครบตามจำนวนที่ต้องการ แต่ธนาคารจะจ่ายธนบัตรพิเศษดังกล่าวนี้เพียง 2 ฉบับก่อน ส่วนที่เหลือจะให้มารับได้ในราวเดือนพฤศจิกายนหรือธันวาคม 2539 นี้ **และเพื่อให้มีการกระจายได้ทั่วถึง ธนาคารแห่งประเทศไทยก็ได้ขอความร่วมมือจากธนาคารพาณิชย์และคลังจังหวัดต่าง ๆ ให้เป็นผู้จัดจำหน่ายและรับจองดังกล่าว...** อย่างนี้เราจะใช้คอมพิวเตอร์เข้าช่วยอย่างไรได้บ้าง?

ผมเชื่อว่าหลาย ๆ คนคงจะเสนอให้ทำออนไลน์เข้ามาที่ส่วนกลาง มันจะเป็นระบบที่พัฒนาได้ง่ายมาก **เพียงแต่ให้โปรแกรมรับชื่อหรือหมายเลขบัตรประจำตัวประชาชนเข้ามาแล้วเช็คกับในไฟล์ที่ส่วนกลางว่าชื่อนี้หรือหมายเลขประจำตัวอันนี้มีอยู่ในไฟล์แล้วหรือไม่ ถ้าไม่มีก็ให้ซื้อได้ แต่ถ้ามีอยู่แล้วก็แสดงว่าได้ใช้สิทธิ์ไปแล้ว เครื่องก็จะปฏิเสธไป** บางคนอาจคุยฟุ้งไปเลยว่าใช้ภาษาดีเบสแค่ 5-6 บรรทัดก็เสร็จแล้ว

โจทย์แบบนี้ดูเหมือนว่าจะไม่ต้องใช้ความรู้เรื่องการจัดเรียงข้อมูล เราคิดง่าย ๆ ว่าชื่อหรือหมายเลขประจำตัวนั้นเป็นสิ่งที่ไม่ซ้ำกันอยู่ และเครื่องสามารถจะให้เราเรียกดูหรืออัพเดตโดยใช้สิ่งเหล่านี้เป็นคีย์ในการเข้าถึงข้อมูลโดยตรง เราจึงไม่ต้องสนใจว่าใครจะซื้อก่อนหรือซื้อทีหลัง ถ้ามีการขอซื้อซ้ำก็จะจับได้

แต่... ระบบใช้พีซีหรือพีซีแลน กับภาษาดีเบสจะทำงานนี้ได้หรือ? ธนบัตรทั้งหมด 1 ล้านฉบับ ซื้อได้ไม่เกินคนละ 5 ฉบับ ก็จะต้องมีคนซื้อได้ไม่น้อยกว่า 2 แสนคน และถ้ามีการแย่งกันซื้อก็คงจะขายหมดใน 2-3 วัน เฉลี่ยจึงตกเข้าไปวันละ 1 แสนรายการ เครื่องเล็ก ๆ อย่างนี้ทำไม่ไหวแน่... ถึงตรงนี้ก็จะเริ่มนึกถึงเครื่องระดับมินิและเมนเฟรม แต่พอพูดถึงเครื่องอย่างนั้นก็จะมีคำถามว่าจะเอาอะไรเป็นเทอร์มินัล จะขอสายสื่อสารไปทั่วประเทศได้อย่างไร ใครจะเป็นคนวิ่งไปติดตั้งและฝึกอบรมผู้ใช้เทอร์มินัล จะออกรายงานเพื่อสรุปยอดรวมเงินที่จะส่งเข้ามาทางกทม. ได้อย่างไร จะออกหลักฐานอะไรให้ผู้ซื้อได้บ้าง การติดต่อเพื่อมารับส่วนที่เหลือได้อย่างไร.... ฯลฯ ..... ข้อสรุปคือ..... มันทำไม่ได้หรอกครับ และมันก็ไม่คุ้มค่าที่จะพัฒนาระบบงานเพื่อใช้แค่ 2-3 วันเท่านั้น

ขอมาดูตัวอย่างที่ 2 กันบ้าง เรื่องนี้เป็นเรื่องที่ต่อเนื่องมาจากข้างต้น คือหลังจากเปิดจำหน่ายไปแล้วก็จะมีคำถามว่า เราจะส่งธนบัตรที่ระลึกจำนวน 3 ฉบับที่จะมาในเดือนพฤศจิกายนหรือธันวาคมนั้นได้อย่างไร? เรื่องอย่างนี้นักคอมพิวเตอร์มักจะบอกรับทันที พวกเรามักจะคิดว่าเป็นเรื่องง่าย คือให้จุดจำหน่ายต่าง ๆ จดบันทึกการจองแล้วส่งข้อมูลเข้ามาที่ส่วนกลาง ส่วนกลางจะเจาะข้อมูลแล้วนำเข้าเครื่อง เมื่อได้แล้วก็จัดเรียงข้อมูลและพิมพ์จ่าหน้าซองได้โดยใช้แผ่นสติกเกอร์ที่มีขายทั่วไป ซองสามารถพิมพ์เตรียมเอาไว้ล่วงหน้า เมื่อตัวธนบัตรที่ระลึกส่งมาถึงมือเราก็จัดใส่ซองและส่งออกไปได้เลย

ข้อเสนอแบบนี้จะโดนทักท้วงตั้งแต่แรก คือมีคำถามว่าส่งของมีค่าอย่างนี้ทางไปรษณีย์ได้อย่างไร? ถ้าของเขาหายไปจะเอาอะไรไปแสดงว่าได้ส่งไปแล้ว?

นักคอมพิวเตอร์ก็จะคิดได้อย่างฉับไวว่าไม่ต้องส่งตัวธนบัตรออกไป ให้ส่งแค่เป็นใบเตือนว่าธนบัตรได้มาถึงเมืองไทยแล้ว และให้ผู้มีชื่อตามที่แจ้งมานี้ไปรับได้ที่ธนาคารแห่งประเทศไทย

โดนเฉ่งกลับมาอีกอย่างแน่นอน คน 2 แสนกว่าคนจะมาเบียดกันที่หน้าธนาคารแห่งประเทศไทยเพื่อมารับธนบัตรที่ระลึกในวันเดียวกัน เรื่องนี้ทำไม่ได้ ครั้งก่อนโน้นเคยมีการจำหน่ายธนบัตรพิเศษราคา 60 บาทมาครั้งหนึ่งแล้ว ผู้คนมาขอแลกซื้อกันเป็นหมื่น ต้องเข้าคิวยาวเป็นกิโล (เมตร) ผู้คนเบียดเสียดกันและเดินย่ำไปบนสนามหญ้าจนพังไปครึ่งหนึ่ง

เมื่อทำแบบนั้นไม่ได้นักคอมพิวเตอร์ก็ต้องเสนอใหม่ ทางออกคือพิมพ์รายชื่อโดยแยกตามจุดจำหน่าย และในแต่ละลิสติงก็ให้จัดเรียงตามชื่อคนหรือหมายเลขประจำตัวประชาชน แล้วส่งกลับไปให้ธนาคารพาณิชย์และคลังจังหวัดชุดเดิมเพื่อใช้ในการแจกจ่ายธนบัตรที่ระลึกส่วนที่เหลือ

ข้อเสนออันนี้ก็พอจะดูเข้าที คือจะเห็นได้ว่ามีประโยชน์ การพิมพ์โดยจัดเรียงตามชื่อหรือหมายเลขประจำตัวนั้นจะช่วยให้ค้นหาได้ง่าย ผู้มารับจะต้องแสดงบัตรประจำตัวประชาชนหรือบอกชื่อ เจ้าพนักงานก็จะไล่ดูตามรายชื่อที่เราพิมพ์ไปให้ แต่อย่างไรก็ตามมันก็ยังมีคำถามออกมาว่า ท่านพอจะประมาณได้ไหมว่าเครื่องจะใช้เวลาพิมพ์รายชื่อทั้งหมดเป็นกี่ชั่วโมง ถ้าท่านใช้ภาษาดีเบส และสั่งให้ select และพิมพ์ตามปกติ ผมบอกได้เลยว่า 3 วัน 3 คืนก็ยังไม่เสร็จ

งานลักษณะข้างต้นนี้ต้องใช้วิธีเรียกเอา routine "sorting" ออกมาใช้งาน เราต้องทำงานแบบ Batch Processing และต้องใช้เครื่องพิมพ์ความเร็วสูง (ระดับ 1200 lines/min ขึ้นไป) การจัดเรียงข้อมูลแบบ Batch Processing หรือที่บางคนเรียกว่า "off-line sorting" เป็นจำนวน 200,000-300,000 เรคอร์ดนั้น จะใช้เวลาประมาณ 10-20 นาที ส่วนการพิมพ์นั้นจะใช้เวลาประมาณ 4 ชั่วโมง ซึ่งอยู่ในวิสัยที่ทำได้

ขอผ่านมาถึงตัวอย่างที่ 3 ตัวอย่างนี้เป็นเรื่องใกล้ตัว คือเรื่องของการทำเคลียริงเช็ค เช็คที่พวกเราสั่งจ่ายให้แก่กันและกันนั้นจะถูกนำไปเข้าฝากที่ธนาคารต่าง ๆ เช็คกว่ากึ่งหนึ่งจะเป็นประเภทที่ผู้จ่ายมีบัญชีอยู่ธนาคารหนึ่ง แต่ผู้รับมีบัญชีอยู่อีกธนาคารหนึ่ง ธนาคารที่รับเช็คเข้ามานั้นมีหน้าที่นำไปเรียกเก็บเงินจากธนาคารผู้ออกเช็ค เรื่องนี้ถ้าให้ขับมอเตอร์ไซด์ไปเรียกเก็บกันทีละใบก็คงวิ่งกันให้วุ่น ทั้งนี้เพราะเช็คประเภทนี้มีถึงวันละ 300,000 ถึง 600,000 ฉบับ วิธีการที่จะลดการวิ่งไปวิ่งมาก็คือให้รวบรวมมาปีกใหญ่ แล้วนัดแนะกันไปพบและแลกเปลี่ยนเช็คกันวันละ 1 หรือ 2 ครั้ง

ศูนย์กลางที่เรานัดแนะไปพบกันคือ "ศูนย์เคลียริง" ในการแลกเปลี่ยนกันนี้ก็จะมีการคิดเงินซึ่งกันและกัน เขาเรียกกิจกรรมนี้ว่า "ทำดุล" การที่เราใช้คำว่า "ดุล" ก็เพราะเมื่อรวมรับและจ่ายทั้งหมดแล้วต้องเท่ากัน คนที่ต้องจ่าย (เพราะถูกเรียกเก็บมาก) นั้นเราเรียกว่า "เสียดุล" ส่วนคนที่จะได้รับเงิน (เพราะส่งเช็คเรียกเก็บเงินได้มากกว่า) นั้นเราเรียกว่า "ได้ดุล" เจ้าตัวเลข "ได้ดุล" และ "เสียดุล" นี้จะเปลี่ยนไปทุก ๆ วัน บางวันอาจได้ดุลหรือเสียดุลหลายพันล้านบาท ซึ่งจะทำให้ธนาคารนั้น ๆ ต้องรีบไปจัดการเรื่องเงินโดยเร็ว พวกนี้จึงต้องการรู้ยอดการได้ดุลหรือเสียดุลโดยเร็ว ซึ่งเราแก้ไขได้โดยให้ธนาคารต่าง ๆ นั้นนำข้อมูลจากเช็คเข้าเครื่องคอมพิวเตอร์เสียก่อน แล้วส่งข้อมูลเข้ามาเป็นแบบ "file transfer" สู่ศูนย์เคลียริง ศูนย์เคลียริงจะได้นำข้อมูลทั้งหมดมาประมวลไว้ล่วงหน้า ส่วนเช็คตัวจริงนั้นจะมาช้าหน่อยก็ไม่เป็นไร ทีนี้ก็จะมาถึงเรื่องของโจทย์ **โจทย์ในงานนี้ก็คือ เมื่อศูนย์เคลียริงรับข้อมูลทางออนไลน์มานั้นควรจะโปรเซสกันอย่างไร?**

คนรุ่นใหม่ก็คิดกันอย่างง่าย ๆ ข้อมูลที่รับมาเป็นไฟล์เล็ก ๆ ย่อย ๆ นั้นรับมาแล้วก็โหลดเข้าฐานข้อมูลไป **และเพื่อประโยชน์ในการแยกไปสู่ธนาคารต่าง ๆ เราก็ให้โหลดโดยมีรหัสธนาคารผู้เรียกเก็บ รหัสสาขาธนาคารผู้เรียกเก็บ รหัสธนาคารผู้ถูกเรียกเก็บ และรหัสสาขาธนาคารผู้ถูกเรียกเก็บเป็นคีย์ในการเรียกดูในภายหลัง** นี่เป็นอย่างขั้นต่ำ จริง ๆ แล้วเรายังโหลดโดยมีคีย์อย่างอื่นอีกมาก เมื่อได้รับข้อมูลจนหมดแล้วเราก็สามารถวิ่งโปรแกรมแบบสอบถาม (Querry) เพื่อหายอดเงินที่แต่ละธนาคารส่งมาเรียกเก็บและยอดเงินที่แต่ละธนาคารต้องจ่ายได้ การออกรายงาน "ดุล" จึงทำได้อย่างสะดวกสบาย (ในฐานะโปรแกรมเมอร์) มันเป็นคำสั่งไม่กี่บรรทัด แต่ท่านทราบไหมว่าโปรเซสทั้งหมดที่เสนอมานี้มันกินเวลากี่ชั่วโมง?

มันใช้เวลาหลายชั่วโมงทีเดียว การโหลดข้อมูลเข้าฐานข้อมูบแบบมีคีย์หลาย ๆ คีย์นั้นกินกำลังเครื่องมหาศาล การออกรายงาน "ดุล" ก็ใช้เวลามากมาย เครื่องคอมพิวเตอร์ขนาด 4 โปรเซสเซอร์ 2 ชุด ยังทำงานแบบไม่ทันจริง ๆ แล้วงานนี้ใช้ระบบแบตช์โปรเซสชิงจะดีกว่านี้แยะ กล่าวคือ ในตอนรับข้อมูลเข้ามาเป็นออนไลน์นั้นก็บันทึกลง sequential file ธรรมดา ๆ เมื่อได้ข้อมูลครบแล้วก็ทำ off-line sorting ด้วยเมมโมรีขนาด 256 เมกะไบต์ เราน่าจะจัดเรียงเสร็จใน 10-20 นาที การออกรายการ "ดุล" ก็เป็นการอ่านแบบ sequential ซึ่งน่าจะเสร็จใน 10-20 นาทีเช่นกัน

สำหรับวันนี้ผมขอจบเพียงแค่นี้ ที่ผมอยากให้พวกเรานักคอมพิวเตอร์ได้คิดก็คือ อย่ามองข้ามความสามารถในการจัดเรียงข้อมูลแบบ off-line ของเครื่องคอมพิวเตอร์

ในชีวิตประจำวันของเราผูกพันกับตัวเลขจนแงะไม่ออก เช้า ๆ ตื่นขึ้นมาก็ต้องดูเวลาว่ากี่โมงแล้ว ยิ่งตอนอยู่ที่ทำงาน ก็ยิ่งต้องคิดคำนวณ ติดต่อคนโน้นคนนี้ให้วุ่นไปหมด พูดถึงการติดต่อ หากเบอร์โทรศัพท์ของสถานที่นั้น ๆ เป็นเบอร์ที่มีหมายเลขคล้องจองกัน หรือเป็นหมายเลขที่สะดุดตา น่าสนใจ เราก็มักจะจำง่าย ดังนั้นบริษัทต่าง ๆ จึงพยายามที่จะมีหมายเลขเหล่านั้น และเรื่องราวที่จะเล่าต่อไปนี้ เป็นเบอร์โทรศัพท์และที่อยู่ที่เกี่ยวกับคอมพิวเตอร์ที่น่าสนใจ ดังนี้คือ

หมายเลขโทรศัพท์ 4 ตัวท้ายของ Microsoft คือ 8080
หมายเลขโทรศัพท์ 4 ตัวท้ายของ PC connection คือ 8088
หมายเลขโทรศัพท์ 4 ตัวท้ายของ Win Tech Journal คือ 0386
หมายเลขโทรศัพท์ 4 ตัวท้ายของ Travelling Sofware คือ 8088
ถนนรอบ ๆ Microsoft ชื่อ Disk Drive
Segate ก็มีถนนชื่อ Disk Drive เหมือนกันกับ Microsoft
Microsoft มีทะเลสาปที่เป็นชื่อท้ายของอีเมล์ของบิลล์ เกตส์ ชื่อ Lake Bilge
สำนักงานใหญ่ของบริษัท Apple ตั้งอยู่ที่ 1 Infinite Loop
บริษัท Delta Products ตั้งอยู่ที่ 15392 Assembly Lane
บริษัท MPS Distributors ตั้งอยู่ที่ 1105 Terminal Way